

Multi-weighted Automata and MSO Logic^{*}

Manfred Droste and Vitaly Perevoshchikov^{**}

Universität Leipzig, Institut für Informatik,
04109 Leipzig, Germany
{droste,perev}@informatik.uni-leipzig.de

Abstract. Weighted automata are non-deterministic automata where the transitions are equipped with weights. They can model quantitative aspects of systems like costs or energy consumption. The value of a run can be computed, for example, as the maximum, limit average, or discounted sum of transition weights. In multi-weighted automata, transitions carry several weights and can model, for example, the ratio between rewards and costs, or the efficiency of use of a primary resource under some upper bound constraint on a secondary resource. Here, we introduce a general model for multi-weighted automata as well as a multi-weighted MSO logic. In our main results, we show that this multi-weighted MSO logic and multi-weighted automata are expressively equivalent both for finite and infinite words. The translation process is effective, leading to decidability results for our multi-weighted MSO logic.

Keywords: Multi-priced automata, quantitative logic, average behavior, power series

1 Introduction

Recently, multi-priced timed automata [5, 6, 17, 20] have received much attention for real-time systems. These automata extend priced timed automata by featuring several price parameters. This permits to compute objectives like the optimal ratio between rewards and costs [5, 6], or the optimal consumption of several resources where more than one resource must be restricted [20]. Arising from the model of timed automata, the multi-weighted setting has also attracted much notice for classical non-deterministic automata [1, 3, 16, 18].

The goal of the present paper is to develop a multi-weighted monadic second order (MSO) logic and to show that it is expressively equivalent to multi-weighted automata.

Büchi’s and Elgot’s fundamental theorems [7, 15] established the expressive equivalence of finite automata and MSO logic. Weighted MSO logic with weights taken from an arbitrary semiring was introduced in [10, 11] and it was shown that a fragment of this weighted logic and semiring-weighted automata on finite and infinite words have the same expressive power [11]. Chatterjee, Doyen, and Henzinger [8, 9] investigated weighted automata modeling the average and long-time behavior of systems. The behavior of such automata cannot be described by semiring-weighted automata. In

^{*} The final publication is available at link.springer.com; DOI: 10.1007/978-3-642-38536-0_36

^{**} Partially supported by DFG Graduiertenkolleg 1763 (QuantLA)

[13, 14], valuation monoids were presented to model the quantitative behaviors of these automata. Their logical characterization was given in [14]. In this paper, we establish, both for finite and infinite words, the Büchi-type result for multi-weighted automata; these do not fit into the framework of other weighted automata like semiring automata [2, 12, 19, 22], or even valuation monoid automata [13, 14].

First, we develop a general model for multi-weighted automata which incorporates several multi-weighted settings from the literature. Next, we define a multi-weighted MSO logic by extending the classical MSO logic with constants which could be tuples of weights. The semantics of formulas should be single weights (not tuples of weights). Different from weighted MSO logics over semirings or valuation monoids, this makes it impossible to define the semantics inductively on the structure of an MSO formula. Instead, for finite words, we introduce an intermediate semantics which maps each word to a finite multiset containing tuples of weights. The semantics of a formula is then defined by applying to the multiset semantics an operator which evaluates a multiset to a single value. Our Büchi-type result for multi-weighted automata on finite words is established by reducing it to the corresponding result of [14] for the product valuation monoid of finite multisets.

In the case of infinite words, it is usually not possible to collect all the information about weights of paths in finite multisets. Therefore, we cannot directly reduce the desired result to the proof given in [14] for infinite words. But we can use the result of [14] to translate each multi-weighted formula of our logic into an automaton over the product ω -valuation monoid of multisets, and we show that the weights of transitions in this automaton satisfy certain properties which allow us to translate it into a multi-weighted automaton.

All our automata constructions are effective. Thus, decision problems for multi-weighted logic can be reduced to decision problems of multi-weighted automata. Some of these problems for automata can be solved whereas for others the details still have to be explored.

2 Multi-weighted Automata on Finite Words

The model of *multi-weighted* (or *multi-priced*) automata is an extension of the model of weighted automata over semirings [2, 12, 19, 22] and valuation monoids [13, 14] by featuring several price parameters. In the literature, different situations of the behaviors of multi-weighted automata were considered (cf. [1, 3, 5, 6, 16, 17, 18, 20]) to model the consumption of several resources. For instance, the model of multi-priced timed automata introduced in [5] permits to describe the optimal ratio between accumulated rewards and accumulated costs of transitions. In this section, we introduce a general model to describe the behaviors of multi-weighted automata on finite words.

Consider an automaton in which every transition carries a reward and a cost. For paths of transitions, we are interested in the ratio between accumulated rewards and accumulated costs. The automaton should assign to each word the maximal reward-cost ratio of accepting paths on w . The idea is to model the weights by elements of the set $M = \mathbb{R} \times \mathbb{R}_{\geq 0}$. We use a valuation function $\text{val} : M^+ \rightarrow M$ to associate to each sequence of such weights a single weight in M . Since our automata are nondeterministic

and a word may have several accepting paths, we obtain a multiset of weights of these paths, hence a multiset of elements from M . We use an evaluator function Φ which associates to each multiset of M a single value. The mapping Φ can be considered as a general summation operator. Now we turn to formal definitions.

To cover also the later case of infinite words, we let $\overline{\mathbb{N}} = \mathbb{N} \cup \{\infty\}$. Let M be a set. A *multiset* over M is a mapping $r : M \rightarrow \overline{\mathbb{N}}$. For each $m \in M$, $r(m)$ is the number of copies of m in r . We let $\text{supp}(r) = \{m \in M \mid r(m) \neq 0\}$, the *support* of r . We say that a multiset r is *finite* if $\text{supp}(r)$ is finite and $\infty \notin r(M)$. We denote the collection of all multisets by $\overline{\mathbb{N}}\langle M \rangle$ and the collection of all finite multisets by $\mathbb{N}\langle M \rangle$.

Definition 1. Let K be a set. A K -valuation structure (M, val, Φ) consists of a set M , a valuation function $\text{val} : M^+ \rightarrow M$ with $\text{val}(m) = m$ for all $m \in M$, and an evaluator function $\Phi : \mathbb{N}\langle M \rangle \rightarrow K$.

A *nondeterministic automaton* over an alphabet Σ is a tuple $\mathcal{A} = (Q, I, T, F)$ where Q is a set of *states*, $I, F \subseteq Q$ are sets of *initial* resp. *final states* and $T \subseteq Q \times \Sigma \times Q$ is a *transition relation*. *Finite paths* $\pi = (t_i)_{0 \leq i \leq n}$ of \mathcal{A} are defined as usual as finite sequences of matching transitions, say $t_i = (q_i, a_i, q_{i+1})$. Then we call the word $w = a_0 a_1 \dots a_n \in \Sigma^+$ the *label* of the path π and π a *path on w* . A path is *accepting* if it starts in I and ends in F . We denote the set of all accepting paths of \mathcal{A} on $w \in \Sigma^+$ by $\text{Acc}_{\mathcal{A}}(w)$.

Definition 2. Let Σ be an alphabet, K a set and $\mathcal{M} = (M, \text{val}, \Phi)$ a K -valuation structure. A *multi-weighted automaton* over Σ and \mathcal{M} is a tuple (Q, I, T, F, γ) where (Q, I, T, F) is a nondeterministic automaton and $\gamma : T \rightarrow M$.

Let \mathcal{A} be a multi-weighted automaton over Σ and \mathcal{M} , $w \in \Sigma^+$ and $\pi = t_0 \dots t_n$ a path on w . The *weight* of π is defined by $\text{Weight}_{\mathcal{A}}(w) = \text{val}(\gamma(t_i))_{0 \leq i \leq n}$. Let $|\mathcal{A}|(w) \in \mathbb{N}\langle M \rangle$ be the finite multiset containing the weights of all accepting paths in $\text{Acc}_{\mathcal{A}}(w)$. Formally, $|\mathcal{A}|(w)(m) = |\{\pi \in \text{Acc}_{\mathcal{A}}(w) \mid \text{Weight}_{\mathcal{A}}(\pi) = m\}|$ for all $m \in M$. The *behavior* $\|\mathcal{A}\| : \Sigma^+ \rightarrow K$ of \mathcal{A} is defined for all $w \in \Sigma^+$ by $\|\mathcal{A}\|(w) = \Phi(|\mathcal{A}|(w))$.

Note that every weighted automaton over a valuation monoid $(M, +, \text{val}, 0)$ (cf. [13, 14]) can be considered as a multi-weighted automaton over the K -valuation structure (M, val, Φ) with $K = M$ and $\Phi : \mathbb{N}\langle M \rangle \rightarrow M$ defined by $\Phi(r) = \sum(m \mid m \in \text{supp}(r) \text{ and } 1 \leq i \leq r(m))$ (as usual, $\sum \emptyset = 0$). Moreover, multi-weighted automata extend the model of weighted automata over valuation monoids in two directions. First, whereas the weights of transitions in multi-weighted automata are taken from M , the behavior is a mapping with the codomain K where K and M do not necessarily coincide. Second, we resolve the nondeterminism in multi-weighted automata using an evaluator function Φ defined on finite multisets.

Next, we consider several examples how to describe the behavior of multi-weighted automata known from the literature using valuation structures. In each of the three examples below, let Σ be an alphabet, $\mathcal{M} = (M, \text{val}, \Phi)$ a K -valuation structure, and \mathcal{A} a multi-weighted automaton over Σ and \mathcal{M} .

Example 3. Let $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. Let $M = \mathbb{R} \times \mathbb{R}_{\geq 0}$, $K = \overline{\mathbb{R}}$, $\text{val}((x_1, y_1), \dots, (x_k, y_k)) = \left(\sum_{i=1}^k x_i, \sum_{i=1}^k y_i \right)$ be the componentwise sum, and Φ

defined by $\Phi(r) = \max_{(x,y) \in \text{supp}(r)} \frac{x}{y}$ where we put $\frac{x}{0} = \infty$ and $\max(\emptyset) = -\infty$. For instance, for every transition weight $(x, y) \in M$, x might mean the reward and y the cost of the transition. Then $\|\mathcal{A}\|(w)$ is the maximal ratio between accumulated rewards and costs of accepting paths on w . The ratio setting was considered first for multi-priced timed automata [5, 6] and also for nondeterministic automata [3, 18].

Example 4. Let $M = \mathbb{R} \times \mathbb{R}$, $K = \mathbb{R} \cup \{\infty\}$ and $p \in \mathbb{R}$. Let val be as in the previous example and $\Phi(r) = \min\{x \mid (x, y) \in \text{supp}(r) \text{ and } y \leq p\}$, for $r \in \mathbb{N}\langle M \rangle$, with $\min(\emptyset) = \infty$. Let t be a transition and $\gamma(t) = (x, y)$. We call x the primary and y the secondary cost. Then $\|\mathcal{A}\|(w)$ is the cheapest primary cost of reaching with w some final state under the given upper bound constraint $p \in \mathbb{R}$ on the secondary cost. The optimal conditional reachability problem for multi-priced timed automata was studied in [20].

Example 5. Let $M = \mathbb{R}^n$ for some $n \geq 1$, $K = \mathbb{R}$, and val be the component-wise sum of vectors. We define $\Phi : \mathbb{N}\langle M \rangle \rightarrow \mathbb{R}$ as follows. Let $r \in \mathbb{N}\langle M \rangle$ and $S = \text{supp}(r)$. Then $\Phi(r) = 0$ if $S = \emptyset$ and $\Phi(r) = \frac{\sum_{v \in S} r(v) \cdot \|v\|}{\sum_{v \in S} r(v)}$ otherwise. Here, for $v = (v_1, \dots, v_n)$, $\|v\| = \sqrt{v_1^2 + \dots + v_n^2}$ is the length of v . Suppose that \mathcal{A} controls the movement of some object in \mathbb{R}^n and each transition t carries the coordinates of the displacement vector of this object. Then, $\|\mathcal{A}\|(w)$ is the value of the average displacement of the object after executing w .

3 Multi-weighted MSO Logic on Finite Words

In this section, we wish to develop a multi-weighted MSO logic where the weight constants are elements of a set M . Again, if weight constants are *pairs* of a reward and a cost, the semantics of formulas must reflect the maximal reward-cost ratio setting, so the weights of formulas should be *single weights*. Then, there arises a problem to define the semantics function inductively on the structure of a formula as in [11, 14]. We solve this problem in the following way. We associate to each word a multiset of elements of M . Here, for disjunction and existential quantification, we use the multiset union. For conjunction, we extend a product operation given on the set M to the Cauchy product of multisets. Similarly, for universal quantification, we extend the valuation function on M^+ to $\mathbb{N}\langle M \rangle^+$. Then, we use an evaluator function Φ which associates to each multiset of elements a single value (e.g. the maximal reward-cost ratio of pairs contained in a multiset).

As in the case of weighted MSO logics over product valuation monoids [14], we extend a valuation structure (cf. Definition 1) with a unit element and a binary operation in order to define the semantics of atomic formulas and of the conjunction.

Definition 6. Let K be a set. A product K -valuation structure (K -pv-structure) $(M, \text{val}, \diamond, \mathbb{1}, \Phi)$ consists of a K -valuation structure (M, val, Φ) , a constant $\mathbb{1} \in M$ with $\text{val}(m\mathbb{1} \dots \mathbb{1}) = m$ for $m \in M$, and a multiplication $\diamond : M \times M \rightarrow M$ such that $m \diamond \mathbb{1} = \mathbb{1} \diamond m = m$ for all $m \in M$.

For the rest of this section, we fix an alphabet Σ and a K -pv-structure $\mathcal{M} = (M, \text{val}, \diamond, \mathbb{1}, \Phi)$. Let V be a countable set of first and second order variables. Lower-case letters like x, y denote first order variables whereas capital letters like X, Y etc. denote second order variables. The syntax of *multi-weighted MSO logic* over Σ and \mathcal{M} is defined as in [4] by the grammar:

$$\begin{aligned}\beta &::= P_a(x) \mid x \leq y \mid x \in X \mid \neg\beta \mid \beta \wedge \beta \mid \forall x\beta \mid \forall X\beta \\ \varphi &::= m \mid \beta \mid \varphi \vee \varphi \mid \varphi \wedge \varphi \mid \exists x\varphi \mid \forall x\varphi \mid \exists X\varphi\end{aligned}$$

where $a \in \Sigma$, $m \in M$, $x, y, X \in V$. The formulas β are called *boolean* formulas and the formulas φ *multi-weighted MSO-formulas*. Note that negation and universal second order quantification are allowed in boolean formulas only. Note also that the boolean formulas have the same expressive power as (unweighted) MSO logic.

The class of *almost boolean* formulas over Σ and M is the smallest class containing all constants $m \in M$ and all boolean formulas and which is closed under \wedge and \vee . A multi-weighted MSO formula φ is *syntactically restricted* if whenever it contains a subformula $\forall x\psi$, then ψ is almost boolean, and if for every subformula $\varphi_1 \wedge \varphi_2$ of φ either both φ_1 and φ_2 are almost boolean, or φ_1 or φ_2 is boolean.

The set $\text{Free}(\varphi)$ of free variables in φ is defined as usual. For $w \in \Sigma^+$, let $\text{dom}(w) = \{0, \dots, |w|-1\}$. Let \mathcal{V} be a finite set of variables with $\text{Free}(\varphi) \subseteq \mathcal{V}$. A (\mathcal{V}, w) -assignment is a mapping $\sigma : \mathcal{V} \rightarrow \text{dom}(w) \cup 2^{\text{dom}(w)}$ where every first order variable is mapped to an element of $\text{dom}(w)$ and every second order variable to a subset of $\text{dom}(w)$. The update $\sigma[x/i]$ for $i \in \text{dom}(w)$ is defined as: $\sigma[x/i](x) = i$ and $\sigma[x/i]|_{\mathcal{V} \setminus \{x\}} = \sigma|_{\mathcal{V} \setminus \{x\}}$. The update for second order variables can be defined similarly. Each pair (w, σ) of a word and (\mathcal{V}, w) -assignment can be encoded as a word over the extended alphabet $\Sigma_{\mathcal{V}} = \Sigma \times \{0, 1\}^{\mathcal{V}}$. Note that a word $(w, \sigma) \in \Sigma_{\mathcal{V}}^+$ represents an assignment if and only if, for every first order variable in \mathcal{V} , the corresponding row in the extended word contains exactly one 1; then (w, σ) is called *valid*. The set of all valid words in $\Sigma_{\mathcal{V}}^+$ is denoted by $\mathcal{N}_{\mathcal{V}}$. We also denote by Σ_{φ} the alphabet $\Sigma_{\text{Free}(\varphi)}$.

Consider again the collection $\mathbb{N}\langle M \rangle$ of all finite multisets over M . Here, we consider the set of natural numbers as the semiring $(\mathbb{N}, +, \cdot, 0, 1)$ where $+$ and \cdot are usual addition and multiplication. The *union* $(r_1 \oplus r_2) \in \mathbb{N}\langle M \rangle$ of finite multisets $r_1, r_2 \in \mathbb{N}\langle M \rangle$ is defined by $(r_1 \oplus r_2)(m) = r_1(m) + r_2(m)$ for all $m \in M$. We define the *Cauchy product* $(r_1 \cdot r_2) \in \mathbb{N}\langle M \rangle$ of two finite multisets $r_1, r_2 \in \mathbb{N}\langle M \rangle$ by

$$(r_1 \cdot r_2)(m) = \sum (r_1(m_1) \cdot r_2(m_2) \mid m_1, m_2 \in M, m_1 \diamond m_2 = m).$$

Note that in the equation above there are finitely many non-zero summands, because the multisets r_1 and r_2 are finite. Let $n \geq 1$ and $r_1, \dots, r_n \in \mathbb{N}\langle M \rangle$. We also define the *valuation* $\text{val}(r_1, \dots, r_n) \in \mathbb{N}\langle M \rangle$ by

$$\text{val}(r_1, \dots, r_n)(m) = \sum \left(\prod_{i=1}^n r_i(m_i) \mid m_1, \dots, m_n \in M, \text{val}(m_1, \dots, m_n) = m \right).$$

Note that the right side of the equation above also contains only finitely many non-zero summands. The *empty multiset* ε is the finite multiset whose support is empty. A *simple multiset* over M is a finite multiset $r \in \mathbb{N}\langle M \rangle$ such that $\text{supp}(r) = \{m_r\}$ and

$\langle m \rangle_{\mathcal{V}}(w, \sigma) = [m]$	
$\langle P_a(x) \rangle_{\mathcal{V}}(w, \sigma) = \begin{cases} [1], & \text{if } w_{\sigma(x)} = a, \\ \varepsilon, & \text{otherwise} \end{cases}$	$\langle \varphi_1 \vee \varphi_2 \rangle_{\mathcal{V}}(w, \sigma) = \langle \varphi_1 \rangle_{\mathcal{V}}(w, \sigma) \oplus \langle \varphi_2 \rangle_{\mathcal{V}}(w, \sigma)$
$\langle x \leq y \rangle_{\mathcal{V}}(w, \sigma) = \begin{cases} [1], & \text{if } \sigma(x) \leq \sigma(y), \\ \varepsilon, & \text{otherwise} \end{cases}$	$\langle \varphi_1 \wedge \varphi_2 \rangle_{\mathcal{V}}(w, \sigma) = \langle \varphi_1 \rangle_{\mathcal{V}}(w, \sigma) \cdot \langle \varphi_2 \rangle_{\mathcal{V}}(w, \sigma)$
$\langle x \in X \rangle_{\mathcal{V}}(w, \sigma) = \begin{cases} [1], & \text{if } \sigma(x) \in \sigma(X), \\ \varepsilon, & \text{otherwise} \end{cases}$	$\langle \exists x \varphi \rangle_{\mathcal{V}}(w, \sigma) = \bigoplus_{i \in \text{dom}(w)} \langle \varphi \rangle_{\mathcal{V} \cup \{x\}}(w, \sigma[x/i])$
$\langle \neg \beta \rangle_{\mathcal{V}}(w, \sigma) = \begin{cases} [1], & \text{if } \langle \beta \rangle_{\mathcal{V}}(w, \sigma) = \varepsilon, \\ \varepsilon, & \text{otherwise} \end{cases}$	$\langle \exists X \varphi \rangle_{\mathcal{V}}(w, \sigma) = \bigoplus_{I \subseteq \text{dom}(w)} \langle \varphi \rangle_{\mathcal{V} \cup \{X\}}(w, \sigma[X/I])$
	$\langle \forall x \varphi \rangle_{\mathcal{V}}(w, \sigma) = \text{val}(\langle \varphi \rangle_{\mathcal{V} \cup \{x\}}(w, \sigma[x/i]))_{i \in \text{dom}(w)}$
	$\langle \forall X \beta \rangle_{\mathcal{V}}(w, \sigma) = \text{val}(\langle \beta \rangle_{\mathcal{V} \cup \{X\}}(w, \sigma[X/I]))_{I \subseteq \text{dom}(w)}$

Table 1. The auxiliary multiset semantics of multi-weighted MSO formulas over a pv-structure

$r(m_r) = 1$, so $r(m) = 0$ for all $m \neq m_r$. We denote such a simple multiset r by $[m_r]$. The collection of all simple multisets over M is denoted by $\text{Mon}(M)$.

As opposed to the case of pv-monoids [14], the pv-structure \mathcal{M} does not contain a commutative and associative sum operation to define the semantics of the disjunction and the existential quantification. For this, we employ the sum of multisets. Let φ be a multi-weighted formula over Σ and \mathcal{M} , and $\mathcal{V} \supseteq \text{Free}(\varphi)$. We define the auxiliary multiset semantics function $\langle \varphi \rangle_{\mathcal{V}} : \Sigma_{\mathcal{V}}^+ \rightarrow \mathbb{N}\langle M \rangle$ relying on the ideas of [11] (cf. also [14]) as follows: for all $(w, \sigma) \notin \mathcal{N}_{\mathcal{V}}$, $\langle \varphi \rangle_{\mathcal{V}}(w, \sigma) = \varepsilon$ and, for all $(w, \sigma) \in \mathcal{N}_{\mathcal{V}}$, $\langle \varphi \rangle_{\mathcal{V}}(w, \sigma)$ is defined inductively as shown in Table 1. Here, $x, y, X \in \mathcal{V}$, $a \in \Sigma$, $m \in M$, β is a boolean formula and $\varphi, \varphi_1, \varphi_2$ are multi-weighted formulas. In Table 1, for the semantics of $\forall X \varphi$ the subsets $I \subseteq \text{dom}(w)$ are enumerated in some fixed order, e.g. lexicographically. For a formula φ , we put $\langle \varphi \rangle = \langle \varphi \rangle_{\text{Free}(\varphi)}$. Then, we define the semantics $\llbracket \varphi \rrbracket : \Sigma_{\varphi}^+ \rightarrow K$ as the composition $\llbracket \varphi \rrbracket = \Phi \circ \langle \varphi \rangle$.

Example 7. Let A be an object on the plane whose displacement is managed by two types of commands: \leftrightarrow and \updownarrow . After receiving the command \leftrightarrow the object moves one step to the left or to the right; after receiving \updownarrow one step up or down. Consider the \mathbb{R} -valuation structure $(\mathbb{R}^2, \text{val}, \Phi)$ from Example 5. We define \diamond as the componentwise sum of vectors and put $\mathbb{1} = (0, 0)$. Then, $\mathcal{M} = (\mathbb{R}^2, \text{val}, \diamond, \mathbb{1}, \Phi)$ is an \mathbb{R} -pv-structure. Consider the following multi-weighted MSO sentence over the alphabet $\Sigma = \{\leftrightarrow, \updownarrow\}$ and the \mathbb{R} -pv-structure \mathcal{M} :

$$\varphi = \forall x((P_{\leftrightarrow}(x) \rightarrow ((-1, 0) \vee (1, 0))) \wedge (P_{\updownarrow}(x) \rightarrow ((0, -1) \vee (0, 1))))$$

where, for a boolean formula φ and a multi-weighted formula ψ , $\beta \rightarrow \psi$ is an abbreviation for $(\beta \wedge \psi) \vee \neg \beta$. For every sequence of commands $w \in \Sigma^+$, the multiset $\langle \varphi \rangle(w)$ contains all possible displacement vectors of the object. For example, let $w = \leftrightarrow \leftrightarrow$. The object has 4 possibilities to move: 1) two steps to the right; 2) one step to the right and then to the home position; 3) one step to the left and then to the home position; 4) two steps to the left. Then $\langle \varphi \rangle(w) = [(2, 0), (0, 0), (0, 0), (-2, 0)]$. The average displacement of the object is given by $\llbracket \varphi \rrbracket$ for each sequence of commands w . For example, $\llbracket \varphi \rrbracket(\leftrightarrow \leftrightarrow) = 1$, $\llbracket \varphi \rrbracket(\leftrightarrow \updownarrow) = \sqrt{2}$.

Note that the multi-weighted MSO logic over K -pv-structures contains the case of weighted MSO logic over semirings (cf. [10, 11]). Hence, in general, multi-weighted MSO logic is expressively more powerful than multi-weighted automata.

Our main result for finite words is the following theorem.

Theorem 8. *Let Σ be an alphabet, K a set, $\mathcal{M} = (M, \text{val}, \diamond, \mathbb{1}, \Phi)$ a K -pv-structure and $s: \Sigma^+ \rightarrow K$. Then $s = \|\mathcal{A}\|$ for some multi-weighted automaton \mathcal{A} over Σ and \mathcal{M} iff $s = \langle\langle\varphi\rangle\rangle$ for a syntactically restricted multi-weighted MSO sentence φ over Σ and \mathcal{M} .*

The proof is similar to the proof of the corresponding Theorem 17 for infinite words. For lack of space, we skip it.

We consider examples of decision problems for multi-weighted MSO logic.

Example 9. Let Σ be an alphabet and $\mathcal{M} = (\mathbb{Q} \times \mathbb{Q}_{\geq 0}, \text{val}, \diamond, (0, 0), \Phi)$ the \mathbb{R} -pv-structure where \diamond is the componentwise sum, and val and Φ are defined as in Example 3. Let φ be a multi-weighted MSO sentence over Σ and \mathcal{M} , and $\nu \in \mathbb{Q}$ a threshold. The $\geq \nu$ -emptiness problem is whether there exists a word $w \in \Sigma^+$ such that $\langle\langle\varphi\rangle\rangle(w) \geq \nu$. If φ is syntactically restricted, then, using our Theorem 8, we can effectively translate φ into a multi-weighted automaton over Σ and \mathcal{M} . Then $\geq \nu$ -emptiness for these multi-weighted automata can be decided in the following way. First, we use a shortest path algorithm to decide whether there exists a path with cost 0, i.e. $\|\mathcal{A}\|(w) = \infty \geq \nu$ for some w . If this is not the case (i.e. the costs of all accepting paths in \mathcal{A} are strictly positive), we use the same technique as for the $\geq \nu$ -emptiness problem for ratio automata with strictly positive costs (cf. [18], Theorem 3). We replace the weight (r, c) of every transition by the single value $r - \nu c$ and obtain a weighted automaton \mathcal{A}' over the max-plus semiring $\mathbb{Q} \cup \{-\infty\}$. Then, $\|\mathcal{A}\|(w) \geq \nu$ iff the semiring-behavior of \mathcal{A}' on w is not less than zero. Then, the decidability of our problem follows from the decidability of the ≥ 0 -emptiness problem for max-plus automata.

Example 10. Let Σ be an alphabet and $\mathcal{M} = (\mathbb{Q}^2, \text{val}, \diamond, (0, 0), \Phi)$ where \diamond is the componentwise sum, and val and Φ are as in Example 4. Again, using our Theorem 8, we can reduce the $\leq \nu$ -emptiness problem (defined similarly as in Example 9) for syntactically restricted multi-weighted MSO logic over Σ and \mathcal{M} to the emptiness problem for multi-weighted automata. This problem is decidable, since the optimal conditional reachability for multi-priced timed automata is decidable [20].

4 Multi-weighted Automata and MSO Logic on Infinite Words

In this section, we develop a general model for both multi-weighted automata and MSO logic on infinite words. Recall that, for a set M , $\overline{\mathbb{N}}\langle\langle M \rangle\rangle$ is the collection of all multisets over M . Let M^ω denote the set of all ω -infinite words over M .

Definition 11. *Let K be a set. A product K - ω -valuation structure (K - ω -pv structure) is a tuple $(M, \text{val}^\omega, \diamond, \mathbb{1}, \Phi)$ where*

- M is a set, $\mathbb{1} \in M$ and $\Phi: \overline{\mathbb{N}}\langle\langle M \rangle\rangle \rightarrow K$;

- $\text{val}^\omega : M^\omega \rightarrow M$ with $\text{val}^\omega(m\mathbb{1}^\omega) = m$ for all $m \in M$;
- $\diamond : M \times M \rightarrow M$ such that $m \diamond \mathbb{1} = \mathbb{1} \diamond m = m$ for all $m \in M$.

A *Muller automaton* over an alphabet Σ is a tuple $\mathcal{A} = (Q, I, T, \mathcal{F})$ where Q is a set of states, $I \subseteq Q$ is a set of initial states, $T \subseteq Q \times \Sigma \times Q$ is a transition relation and $\mathcal{F} \subseteq 2^Q$ is a Muller acceptance condition. *Infinite paths* $\pi = (t_i)_{i \in \omega}$ of \mathcal{A} are defined as infinite sequences of matching transitions, say $t_i = (q_i, a_i, q_{i+1})$. Then we call the word $w = (a_i)_{i \in \omega}$ the *label* of the path π and π a path on w . We say that a path $\pi = (q_i, a_i, q_{i+1})_{i \in \omega}$ is *accepting* if $q_0 \in I$ and $\{q \in Q \mid q = q_i \text{ for infinitely many } i \in \omega\} \in \mathcal{F}$. Let $\text{Acc}_{\mathcal{A}}(w)$ denote the set of all accepting paths of \mathcal{A} on w .

For the rest of this section, we fix an alphabet Σ and a K - ω -pv structure $\mathcal{M} = (M, \text{val}^\omega, \diamond, \mathbb{1}, \Phi)$.

Definition 12. A multi-weighted Muller automaton over Σ and \mathcal{M} is a tuple $\mathcal{A} = (Q, I, T, \mathcal{F}, \gamma)$ where (Q, I, T, \mathcal{F}) is a Muller automaton and $\gamma : T \rightarrow M$.

Let \mathcal{A} be a multi-weighted Muller automaton over Σ and \mathcal{M} , $w \in \Sigma^\omega$ and $\pi = (t_i)_{i \in \omega}$ an accepting path on w . The *weight* of π is defined by $\text{Weight}_{\mathcal{A}}(\pi) = \text{val}^\omega(\gamma(t_i))_{i \in \omega}$. Let $|\mathcal{A}|(w) \in \overline{\mathbb{N}}\langle\langle M \rangle\rangle$ be the multiset containing the weights of paths in $\text{Acc}_{\mathcal{A}}(w)$. Formally, $|\mathcal{A}|(w)(m) = |\{\pi \in \text{Acc}_{\mathcal{A}}(w) \mid \text{Weight}_{\mathcal{A}}(\pi) = m\}|$ where, for an infinite set X , we put $|X| = \infty$. The *behavior* of \mathcal{A} is the ω -series $\|\mathcal{A}\| : \Sigma^\omega \rightarrow K$ defined by $\|\mathcal{A}\|(w) = \Phi(|\mathcal{A}|(w))$.

Remark 13. The multiplication \diamond , the unital element $\mathbb{1}$ and the condition $\text{val}^\omega(m\mathbb{1}^\omega) = m$ are irrelevant for the definition of the behaviors of multi-weighted automata. However, they will be used to describe the semantics of multi-weighted MSO formulas.

We consider several examples of multi-weighted automata \mathcal{A} over Σ and \mathcal{M} , and their behaviors.

Example 14. Consider the reward-cost ratio setting of Example 3 for infinite words. For a sequence $(r_i, c_i)_{i \in \omega} \in (\mathbb{R} \times \mathbb{R}_{\geq 0})^\omega$ of reward-cost pairs, the *supremum ratio* (cf. [5]) is defined by $\limsup_{n \rightarrow \infty} \frac{\sum_{i=0}^n r_i}{\sum_{i=0}^n c_i} \in \overline{\mathbb{R}}$ where $\frac{r}{0} = \infty$. Unfortunately, since $\sum_{i=0}^\infty r_i$ and $\sum_{i=0}^\infty c_i$ may not exist or may be infinite, we cannot proceed as for finite words by considering pairs of accumulated rewards and costs and their ratios. Instead, we can define \mathcal{M} as follows. Let $M = \overline{\mathbb{R}} \times \mathbb{R}_{\geq 0}$, $K = \overline{\mathbb{R}}$ and $\mathbb{1} = (0, 0)$. Let $\mu = (r_i, c_i)_{i \in \omega} \in (\mathbb{R} \times \mathbb{R}_{\geq 0})^\omega$. If $\sum_{i=0}^\infty r_i$ and $\sum_{i=0}^\infty c_i$ are finite, then we put $\text{val}^\omega(\mu) = (\sum_{i=0}^\infty r_i, \sum_{i=0}^\infty c_i)$. Otherwise, we put $\text{val}^\omega(\mu) = \left(\limsup_{n \rightarrow \infty} \frac{\sum_{i=0}^n r_i}{\sum_{i=0}^n c_i}, 1 \right)$. For sequences $\mu \in M^\omega \setminus (\mathbb{R} \times \mathbb{R}_{\geq 0})^\omega$, we define $\text{val}^\omega(\mu)$ arbitrarily keeping $\text{val}^\omega(m\mathbb{1}^\omega) = m$. Let also \diamond be the componentwise sum where $\infty + (-\infty)$ is defined arbitrarily. The evaluator function Φ is defined by $\Phi(r) = \sup_{(x,y) \in \text{supp}(r)} \frac{x}{y}$. Then, $\|\mathcal{A}\|(w)$ is the maximal supremum ratio of accepting paths of w . The corresponding model for timed automata was considered in [5, 6].

Example 15. Let $E_{\max} = (E_{\max}^1, \dots, E_{\max}^n) \in \mathbb{Z}^n$ where $E_{\max}^i > 0$ for all i , and $M = [-E_{\max}, E_{\max}] \subseteq \mathbb{Z}^n$, i.e. M consists of all vectors $(v^1, \dots, v^n) \in \mathbb{Z}^n$ such that $-E_{\max}^i \leq v^i \leq E_{\max}^i$ for each $i \in \{1, \dots, n\}$. Let $K = \mathbb{B} = \{\text{false}, \text{true}\}$, the boolean semiring and $\mathbb{1} = (0, \dots, 0)$. For $u_1 = (u_1^1, \dots, u_1^n)$ and $u_2 = (u_2^1, \dots, u_2^n) \in M$, we put $u_1 \diamond u_2 = (v^1, \dots, v^n)$ where $v^i = \max\{\min\{u_1^i + u_2^i, E_{\max}^i\}, -E_{\max}^i\}$. For $(m_i)_{i \in \omega} \in M^\omega$ we define the sequence $(v_i)_{i \in \omega}$ in M as follows. We put $v_0 = (0, \dots, 0)$ and $v_{i+1} = v_i \diamond m_i$ for all $i \in \omega$. Then, let $\text{val}^\omega((m_i)_{i \in \omega}) = (x^1, \dots, x^n) \in M$ where $x^j = \inf\{v_i^j \mid i \in \omega\}$ for all $1 \leq j \leq n$. Let Φ be defined by $\Phi(r) = \text{true}$ iff there exists $(m^1, \dots, m^n) \in \text{supp}(r)$ with $m^j \geq 0$ for all $1 \leq j \leq n$. This model corresponds to the one-player energy games considered in [16].

The syntax of the *multi-weighted MSO logic* over Σ and \mathcal{M} is defined exactly as for finite words (cf. Section 3). To define the semantics of this logic, we proceed similarly as for finite words, i.e. by means of the auxiliary multiset semantics. For this, we consider $\overline{\mathbb{N}}$ as the totally complete semiring $(\overline{\mathbb{N}}, +, \cdot, 0, 1)$ (cf. [11]) where $0 \cdot \infty = \infty \cdot 0 = 0$. The sum \oplus and the Cauchy product \cdot for infinite multisets from $\overline{\mathbb{N}}\langle\langle M \rangle\rangle$ are defined as for finite words. The ω -valuation $\text{val}^\omega(r_i)_{i \in \omega}$ for $r_i \in \overline{\mathbb{N}}\langle\langle M \rangle\rangle$ is defined for all $m \in M$ by

$$\text{val}^\omega((r_i)_{i \in \omega})(m) = \sum \left(\prod_{i \in \omega} r_i(m_i) \mid (m_i)_{i \in \omega} \in M^\omega \text{ and } \text{val}^\omega(m_i)_{i \in \omega} = m \right).$$

The *empty multiset* $\varepsilon \in \overline{\mathbb{N}}\langle\langle M \rangle\rangle$ and *simple multisets* $[m] \in \overline{\mathbb{N}}\langle\langle M \rangle\rangle$ (for $m \in M$) are defined in the same way as for finite words. Let $\text{Mon}(M) = \{[m] \mid m \in M\}$.

Let φ be a multi-weighted MSO formula over Σ and \mathcal{M} , and $\mathcal{V} \supseteq \text{Free}(\varphi)$. We define the auxiliary multiset semantics $\langle \varphi \rangle_{\mathcal{V}} : \Sigma_{\mathcal{V}}^\omega \rightarrow \overline{\mathbb{N}}\langle\langle M \rangle\rangle$ inductively on the structure of φ as in Table 1 where we have to replace val by val^ω . For $w \in \Sigma^\omega$, we let $\text{dom}(w) = \omega$. To define the semantics $\langle \forall X \varphi \rangle$, we have to extend val^ω for multisets to index sets of size continuum such that $\text{val}^\omega((r_i)_{i \in I}) = \varepsilon$ whenever $r_i = \varepsilon$ for some $i \in I$, and $\text{val}^\omega(([\mathbb{1}])_{i \in I}) = [\mathbb{1}]$. The *semantics* of φ is defined by $\langle\langle \varphi \rangle\rangle = \Phi \circ \langle \varphi \rangle$.

Example 16. Assume that a bus can operate two routes A and B which start and end at the same place. The route R lasts t_R time units and profits p_R money units on the average per trip, for $R \in \{A, B\}$. We may be interested in making an infinite schedule for this bus which is represented as an infinite sequence from $\{A, B\}^\omega$. This schedule must be fair in the sense that both routes A and B must occur infinitely often in this timetable (even if the route A or B is unprofitable). The optimality of the schedule is also preferred (we wish to profit per time unit as much as possible). We consider the K - ω -pv structure \mathcal{M} from Example 14 and a one-element alphabet $\Sigma = \{\tau\}$ which is irrelevant here. Now we construct a weighted MSO sentence φ over Σ and \mathcal{M} to define the optimal income of the bus per time unit (supremum ratio between rewards and time):

$$\varphi = \exists X \left(\bigoplus_{i \in \omega} x(x \in X) \wedge \bigoplus_{i \in \omega} x(x \notin X) \wedge \forall x ((x \in X \rightarrow (p_A, t_A)) \wedge (x \notin X \rightarrow (p_B, t_B))) \right)$$

where $\bigoplus x\psi$ is an abbreviation for a boolean formula $\forall y (\neg \forall x (\neg (y \leq x \wedge \psi)))$. Here, the second order variable X corresponds to the set of positions in an infinite schedule

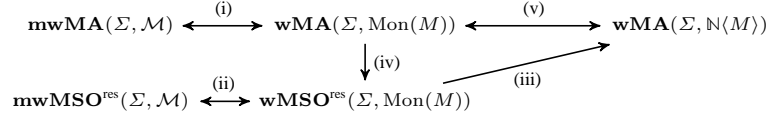


Fig. 1. The proof scheme of Theorem 17

which can be assigned to the route A. Then,

$$|\varphi|(\tau^\omega) = \sup \left\{ \limsup_{n \rightarrow \infty} \frac{p_A \cdot |I \cap \bar{n}| + p_B \cdot |I^c \cap \bar{n}|}{t_A \cdot |I \cap \bar{n}| + t_B \cdot |I^c \cap \bar{n}|} \mid I \subseteq \mathbb{N} \text{ with } I, I^c \text{ infinite} \right\}$$

where $\bar{n} = \{0, \dots, n\}$ and $I^c = \mathbb{N} \setminus I$.

Now we state our main result for infinite words.

Theorem 17. *Let Σ be an alphabet, K a set and $\mathcal{M} = (M, \text{val}^\omega, \diamond, \mathbb{1}, \Phi)$ a K - ω -pv structure. Let $s : \Sigma^\omega \rightarrow K$ be an ω -series. Then $s = \|\mathcal{A}\|$ for some multi-weighted Muller automaton \mathcal{A} over Σ and \mathcal{M} iff $s = \langle\langle\varphi\rangle\rangle$ for some syntactically restricted multi-weighted MSO sentence φ over Σ and \mathcal{M} .*

In the rest of this section, we give the proof idea of this theorem. Let $\mathbf{mwMA}(\Sigma, \mathcal{M})$ denote the collection of all multi-weighted Muller automata over Σ and \mathcal{M} . Let $\mathcal{A} \in \mathbf{mwMA}(\Sigma, \mathcal{M})$. We can consider $|\mathcal{A}|$ as an ω -series $|\mathcal{A}| : \Sigma^\omega \rightarrow \overline{\mathbb{N}}\langle\langle M \rangle\rangle$. We call $|\mathcal{A}|$ the *multiset-behavior* of \mathcal{A} . Then $\|\mathcal{A}\| = \Phi \circ |\mathcal{A}|$. Let $\mathbf{mwMSO}^{\text{res}}(\Sigma, \mathcal{M})$ denote the set of all syntactically restricted multi-weighted MSO sentences over Σ and \mathcal{M} . Since, for any multi-weighted formula φ , $\langle\langle\varphi\rangle\rangle = \Phi \circ \langle\varphi\rangle$, it suffices to show that $\mathbf{mwMA}(\Sigma, \mathcal{M})$ with the multiset-behavior and $\mathbf{mwMSO}^{\text{res}}(\Sigma, \mathcal{M})$ with the multiset-semantics are expressively equivalent.

For this, we can show that $(\overline{\mathbb{N}}\langle\langle M \rangle\rangle, \oplus, \text{val}^\omega, \cdot, \varepsilon, [\mathbb{1}])$ is an ω -pv monoid as defined in [14]. Let $D \subseteq \overline{\mathbb{N}}\langle\langle M \rangle\rangle$. We denote by $\mathbf{wMA}(\Sigma, D)$ the collection of weighted automata over Σ and the ω -pv monoid $\overline{\mathbb{N}}\langle\langle M \rangle\rangle$ where the weights of transitions are taken from D . Let $\mathbf{wMSO}^{\text{res}}(\Sigma, D)$ denote the set of syntactically restricted sentences over Σ and the ω -pv monoid $\overline{\mathbb{N}}\langle\langle M \rangle\rangle$ with constants from D . Let $\llbracket \varphi \rrbracket$ denote the semantics of $\varphi \in \mathbf{wMSO}^{\text{res}}(\Sigma, \mathcal{M})$ as defined in [14]. The proof scheme of our result is depicted in Fig. 1. Here, \leftrightarrow means the expressive equivalence and \rightarrow the expressive inclusion.

- (i) If we replace the weight $m \in M$ of every transition of a multi-weighted automaton \mathcal{A} by the simple multiset $[m] \in \text{Mon}(M)$, we obtain a weighted automaton \mathcal{A}' over the pv monoid $\mathbb{N}\langle M \rangle$ such that the pv-monoid behavior of \mathcal{A}' is equal to $|\mathcal{A}|$. Conversely, we can replace the weights $[m]$ in \mathcal{A}' by m to obtain a multi-weighted automaton with the same behavior.
- (ii) Similarly to (i), we replace the constants m occurring in MSO formulas by simple multisets $[m]$ and vice versa.
- (iii) The proof is based on the proof of Theorem 6.2 (a) of Droste and Meinecke [14]. We proceed inductively on the structure of $\varphi \in \mathbf{wMSO}^{\text{res}}(\Sigma, \text{Mon}(M))$.

Using the property $\text{val}^\omega(m1^\omega) = m$ for $m \in M$, we show that every almost boolean formula is equivalent to a weighted Muller automaton with weights from $\text{Mon}(M) \subseteq \mathbb{N}\langle M \rangle$. Let φ, φ_1 and φ_2 be weighted MSO formulas with constants from $\text{Mon}(M)$ such that $\llbracket \varphi \rrbracket, \llbracket \varphi_1 \rrbracket$ and $\llbracket \varphi_2 \rrbracket$ are recognizable by weighted Muller automata with weights from $D \subseteq \overline{\mathbb{N}}\langle M \rangle$. Let β be any boolean formula. It can be shown that $\llbracket \varphi_1 \vee \varphi_2 \rrbracket, \llbracket \exists x \varphi \rrbracket, \llbracket \exists X \varphi \rrbracket$ and $\llbracket \varphi \wedge \beta \rrbracket = \llbracket \beta \wedge \varphi \rrbracket$ are also recognizable by weighted Muller automata with weights from D . If φ is almost boolean, then $\llbracket \varphi \rrbracket$ is an ω -recognizable step function with coefficients from $\mathbb{N}\langle M \rangle$. Using the construction of Lemma 8.11 of [11], cf. Theorem 6.2 of [14], we establish that $\llbracket \forall x \varphi \rrbracket$ is recognizable by a weighted automaton with weights from $\mathbb{N}\langle M \rangle$.

- (iv) The proof follows from Theorem 6.2 of [14] where a weighted automaton with weights in $D \subseteq \overline{\mathbb{N}}\langle M \rangle$ was translated into an MSO sentence with weights in D .
- (v) Let $\mathcal{A} = (Q, I, T, \mathcal{F}, \gamma) \in \mathbf{wMA}(\Sigma, \mathbb{N}\langle M \rangle)$. We construct an automaton $\mathcal{A}' = (Q', I', T', \mathcal{F}', \gamma') \in \mathbf{wMA}(\Sigma, \text{Mon}(M))$ with the same behavior by unfolding each single transition of \mathcal{A} labeled by a finite multiset into several transitions labeled by elements of this multiset as follows.
 - $Q' = I \cup \{(q, m, i) : t = (p, a, q) \in T, m \in \text{supp}(\gamma(t)), 1 \leq i \leq \gamma(t)(m)\}$
 - $I' = I, \mathcal{F}' = \{\{(q_1, m_1, k_1), \dots, (q_n, m_n, k_n)\} \subseteq Q' \setminus I \mid \{q_1, \dots, q_n\} \in \mathcal{F}\}$.
 - $T' = T_1 \cup T_2$, where T_1 consists of all transitions $(p, a, (q, m, i))$ from $I \times \Sigma \times (Q' \setminus I)$ with $(p, a, q) \in T$; T_2 consists of all transitions $((q_1, m_1, i_1), a, (q_2, m_2, i_2))$ from $(Q' \setminus I) \times \Sigma \times (Q' \setminus I)$ with $(q_1, a, q_2) \in T$.
 - For all $t = (q', a, (q, m, i)) \in T'$, let $\gamma'(t) = [m]$.

5 Conclusion

We have extended the use of weighted MSO logic to a new class of multi-weighted settings. We just note that, as in [14], for K -pv-structures and K - ω -pv structures with additional properties there are larger fragments of multi-weighted MSO logic which are still expressively equivalent to multi-weighted automata. Since our translations from formulas to automata are effective, we can reduce the decidability problems for multi-weighted logics to the corresponding problems for multi-weighted automata. Decidability results of, e.g., [5, 16, 18, 20] lead to decidability results for multi-weighted nondeterministic automata. However, for infinite words, the authors did not consider Muller acceptance condition for automata. Therefore, our future work will investigate decision problems for multi-weighted Muller automata. Also, weighted MSO logic for weighted timed automata was investigated in [21]. In our further work, we wish to combine the ideas of [21] and the current work to obtain a Büchi theorem for multi-weighted timed automata.

References

- [1] S. Bauer, L. Juhl, K. Larsen, A. Legay, and J. Srba. A logic for accumulated-weight reasoning on multiweighted modal automata. In *Proc. of the 6th Int. Symp. on Theoret. Aspects of Software Engineering (TASE'12)*, pages 77–84. IEEE Computer Society Press, 2012.

- [2] J. Berstel and C. Reutenauer. *Rational Series and Their Languages*, volume 12 of *EATCS Monographs on Theoretical Computer Science*. Springer, 1988.
- [3] R. Bloem, K. Greimel, T. A. Henzinger, and B. Jobstmann. Synthesizing robust systems. In *Proc. of 9th Int. Conf. on Formal Methods in Computer-Aided Design (FMCAD'2009)*, pages 85–92. IEEE, 2009.
- [4] B. Bollig and P. Gastin. Weighted versus probabilistic logics. In *Proc. of the 13th Int. Conf. on Developments in Language Theory (DLT'09)*, volume 5583 of *LNCS*, pages 18–38. Springer, 2009.
- [5] P. Bouyer, E. Brinksma, and K. G. Larsen. Staying alive as cheaply as possible. In *Hybrid Systems: Computation and Control, 7th Int. Workshop*, volume 2993 of *LNCS*, pages 203–218. Springer, 2004.
- [6] P. Bouyer, E. Brinksma, and K. G. Larsen. Optimal infinite scheduling for multi-priced timed automata. *Formal Methods in System Design*, 32(1):3–23, 2008.
- [7] J. R. Büchi. Weak second-order arithmetic and finite automata. *Z. Math. Logik und Grundl. Math.*, 6:66–92, 1960.
- [8] K. Chatterjee, L. Doyen, and T. A. Henzinger. Quantitative languages. In *Proc. of the 22nd Int. Workshop on Comp. Sci. Logic (CSL'08)*, volume 5213 of *LNCS*, pages 385–400. Springer, 2008.
- [9] K. Chatterjee, L. Doyen, and T. A. Henzinger. Expressiveness and closure properties for quantitative languages. *Logical Methods in Comp. Sci.*, 6(3), 2010.
- [10] M. Droste and P. Gastin. Weighted automata and weighted logics. *Theoret. Comp. Sci.*, 380(1-2):69–86, 2007.
- [11] M. Droste and P. Gastin. Weighted automata and weighted logics. In Droste et al. [12], chapter 5.
- [12] M. Droste, W. Kuich, and H. Vogler, editors. *Handbook of Weighted Automata*. EATCS Monographs on Theoretical Computer Science. Springer, 2009.
- [13] M. Droste and I. Meinecke. Weighted automata and regular expressions over valuation monoids. *Int. J. Found. Comput. Sci.*, 22(8):1829–1844, 2011.
- [14] M. Droste and I. Meinecke. Weighted automata and weighted MSO logics for average and long-time behaviors. *Inf. Comput.*, 220-221:44–59, 2012.
- [15] C. C. Elgot. Decision problems of finite automata design and related arithmetics. *Trans. Amer. Math. Soc.*, 98:21–51, 1961.
- [16] U. Fahrenberg, L. Juhl, K. G. Larsen, and J. Srba. Energy games in multiweighted automata. In *Proc. of the 8th Int. Conf. on Theoret. Aspects of Computing (ICTAC'11)*, volume 6916 of *LNCS*, pages 95–115. Springer, 2011.
- [17] U. Fahrenberg, K. G. Larsen, and C. R. Thrane. Model-based verification and analysis for real-time systems. In *Software and Systems Safety - Specification and Verification*, volume 30 of *NATO Science for Peace and Security Series - D: Information and Communication Security*, pages 231–259. IOS Press, 2011.
- [18] E. Filiot, R. Gentilini, and J.-F. Raskin. Quantitative languages defined by functional automata. In *Proc. of the 23rd Int. Conf. on Concurrency Theory (CONCUR'12)*, volume 7454 of *LNCS*, pages 132–146. Springer, 2012.
- [19] W. Kuich and A. Salomaa. *Semirings, Automata and Languages*, volume 5 of *EATCS Monographs on Theoretical Computer Science*. Springer, 1986.
- [20] K. G. Larsen and J. I. Rasmussen. Optimal conditional reachability for multi-priced timed automata. In *Proc. of the 8th Int. Conf. on Found. of Software Science and Computation Structures (FOSSACS '05)*, volume 3441 of *LNCS*, pages 234–249. Springer, 2005.
- [21] K. Quaas. MSO logics for weighted timed automata. *Form. Methods Syst. Des.*, 38(3):193–222, 2011.
- [22] A. Salomaa and M. Soittola. *Automata-Theoretic Aspects of Formal Power Series*. Texts and Monographs in Computer Science. Springer, 1978.